AF

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Mail Stop Appeal Brief
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on February 22, 2007
(Date of Deposit)

David M. Lockman
Name of person mailing Document or Fee

Signature

February 22, 2007
Date of Signature

Re:	Application of:	Osman Ahmed
	Serial No.:	10/668,949
	Filed:	September 23, 2003
	For:	System and Method for Developing and Processing Building System Control Solutions
	Group Art Unit:	2168
	Confirmation No.:	3299
	Examiner:	Olubusola Oni
	Our Docket No.:	1867-0008

TRANSMITTAL OF APPEAL BRIEF

Please find for filing in connection with the above patent application the following:

1. Appeal Brief;

2. Check in the amount of $500.00; and

3. One (1) return post card.

Enclosed please find a check in the amount of $500.00 to cover the filing fee of a Brief on Appeal as required by 37 C.F.R. § 1.17(c). Please charge any deficiency, or credit any overpayment to Deposit Account No. 13-0014.

Respectfully Submitted,
MAGINOT, MOORE & BECK

February 22, 2007

David M. Lockman
Registration No. 34,214
Chase Tower
111 Monument Circle, Suite 3250
Indianapolis, Indiana 46204-5109

Enclosures

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re patent application of:	)	MMB Docket No.: **1867-0008**
	)	
Inventor: **Osman Ahmed**	)	
	)	
Application No.: **10/668,949**	)	Examiner: **Olubusola Oni**
	)	
Filed: **September 23, 2003**	)	Group Art No.: **2168**
	)	
For: **System And Method For Developing And Processing Building System Control Solutions**	)	Confirmation No.: **3299**

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner of Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on

February 22, 2007
Date of Deposit

David M. Lockman
Name of person mailing Document or Fee

David M Lockman
Signature of person mailing Document or Fee

February 22, 2007
Date of Signature

BRIEF ON APPEAL

Hon. Commissioner of Patents and Trademarks

Alexandria, VA 20231

Sir:

This is an appeal under 37 CFR § 41.31 to the Board of Patent Appeals and Interferences of the United States Patent and Trademark Office from the final rejection of the claims 1-28 of the above-identified patent application. These claims were rejected in an Office Action dated September 22, 2006. On December 22, 2006, Applicant filed a Notice of Appeal.

Please provide any extension of time that may be necessary and charge any fees that may be due to Account No. 13-0014, but do not include any payment of issue fees.

(1) REAL PARTY IN INTEREST

The real party in interest is Siemens Corporation, the assignee of record for this patent application.

(2) RELATED APPEALS AND INTERFERENCES

No prior and pending appeals, judicial proceedings, and interferences related to this patent application are known to the applicant.

(3) STATUS OF CLAIMS

Claims 1-28 are pending in the application and claims 29-55 were withdrawn in response to a restriction requirement presented in the Office Action of March 24, 2006.

Claims 1-28 are finally rejected, and are being appealed.

Each of the appealed claims 1-28 is shown in the Claims Appendix attached to this Appeal Brief.

(4) STATUS OF AMENDMENTS

Appellant has filed no amendments subsequent to the Final Office Action mailed September 22, 2006.

(5) SUMMARY OF CLAIMED SUBJECT MATTER

Independent claim 1 is directed to a system for a building system application. The system facilitates the development of application solutions for controlling environmental systems in buildings as well as the monitoring of such systems. The system includes a database, a data provider interface for converting between a common database access method and a database application programming interface (API), and an application infrastructure. *Specification*, page 9, line 10 to page 10, line 10. The application infrastructure includes a system design converter, a computer tool interface, and an external program module. *Specification*, page 20, line 15 to page 22, line 10. The data provider interface converts between a *common* database access method and a database API. As set forth in the specification at page 23, lines 3-13, application programs use a common database access method. This common database access method enables the application programs to generate queries and receive responses with a single access method. Consequently, the application designer is not required to program database queries and parse responses with knowledge of multiple database application programming interfaces. Instead, the data provider performs these tasks.

The system design converter accepts application definition data as input. These data describe components in a building system and the control logic for a building system. *Specification*, page 20, lines 19-23. The system design converter generates computer statements from the application definition data. *Specification*, page 20, line 23 – page 2, line 6. The computer tool interface coupled to the system design converter provides the system design converter with data from the database through the data provider interface. These data are used by the system design converter for the generation of the computer statements. *Specification*, page 21, line 6 to page 22, line 10. The external program module interface is coupled to the system design converter to provide the system design converter with external program modules for the generation of an application solution. *Specification*, page 22, lines 4-10.

The above-described application infrastructure allows the application programmer to develop an application without requiring the programmer to have knowledge of multiple database access methods and their particular APIs. The external program module interface enables the system design converter to incorporate external program modules into the generated application solutions. This feature enables the system designer to use previously tested and validated external programs. Consequently, a building system control application developer may concentrate on those aspects of an application solution that provide efficient control over the building parameters affected by the application without requiring the developer to have extensive knowledge of database access methods and external program interfaces.

Independent claim 15 is directed to a method for supporting a building system application. The method includes converting application definition data into computer statements for implementing the control logic of the definition data, *Specification*, page 20, lines 19-23, converting between common database access method instructions in the computer statements and database API instructions for a database, *Specification*, page 23, lines 3-13, and generating building system applications with the computer statements that incorporate external program modules and data obtained from the database. *Specification*, page 22, lines 4-10. This method facilitates generation of building system control application solutions from application definition data without requiring the developer of the definition data to have extensive knowledge of database access methods and external program interfaces.

(6) GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. The first ground of rejection to be reviewed on appeal is whether U.S. Patent No. 6,141,595 to Gloudeman et al. anticipates the inventions of claims 1-3, 5-17, and 19-28 under 35 U.S.C. § 102(b).

2. The second ground of rejection to be reviewed on appeal is whether U.S. Patent No. 6,141,595 to Gloudeman et al. in view of Publ. No. 2003/0229652 to Bakalash renders claims 4 and18 obvious under 35 U.S.C. § 103(a).

(7) ARGUMENT

I. Rejection under 35 USC 102(b) Over U.S. Patent No. 6,141,595

CLAIMS 1-3, 11-13, 15-17, and 25-27

The Examiner has rejected claims 1-3, 11-13, 15-17, and 25-27 under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 6,141,595 to Gloudeman et al. (hereinafter "Gloudeman"). This ground of rejection is unsupported by the cited reference as it does not teach every aspect of the claimed invention either explicitly or impliedly. *See MPEP 706.02,* ¶ *IV.* Specifically, the Gloudeman reference does not teach, explicitly or impliedly, a data provider interface, a system design converter, a computer tool interface, and an external program module interface as those elements are set forth in the rejected claims. Each of these elements will be addressed below.

A. The Gloudeman reference does not teach "a data provider interface for converting between a common database access method and a database application programming interface (API)." The data provider interface converts between a *common* database access method and a database API. As set forth in the specification at page 23, lines 3-13, application programs use a common database access method. This common database access method enables the application programs to generate queries and receive responses with a single access method. Consequently, the application designer is not required to program database queries and parse responses with knowledge of multiple database application programming interfaces.

The database interface of Gloudeman requires, on the other hand, external systems to communicate with the databases 58, 60, 70, and 72 using the APIs of the databases through the standard communication and database APIs of layer 54, *Gloudeman*, col. 4, line 62 – col. 5, line 2, or one of a plurality of third party access methods that is converted by the third party interface 56, *Gloudeman*, col. 5, lines 2-9. Thus, an external system communicating through layer 54 requires no data provider that performs any conversion as the external system communicates with the database using the API of the database. If an external system does not use the standard communication and database APIs of layer 54, then it may communicate with a non-conforming method that is converted to the appropriate API for the database being accessed. *Gloudeman*, col. 5, lines 2-9. In this case, each non-conforming external system may use one or more different third party access methods. These third party access methods are converted by layer 56, through the use of appropriate protocol converters, for communication with the API for the database being accessed. Thus, Gloudeman does not teach or suggest a common database access method for a variety of databases having different APIs. The absence of a teaching or suggestion in Gloudeman that all non-conforming external systems use a *common* access method for accessing the database that is converted for communication with the API of the database renders Gloudeman ineffective as an anticipatory reference to the invention of claims 1-3, 11-13, 15-17, and 25-27.

B. The Gloudeman reference does not teach, explicitly or impliedly, "a system design converter for converting application definition data into computer statements for implementing control logic of application definition data." The system design converter accepts as input application definition data. These data describe components in a building system and the control logic for a building system. *Specification*, page 20, lines 19-23. The system design converter generates computer statements from the application definition data. *Specification*, page 20, line 23 – page 2, line 6. These computer statements are executed by the controllers of the building system to perform the application program corresponding to the application definition data. Thus, this claim limitation requires a system design converter that generates computer statements from application definition data.

The Gloudeman reference does not teach or suggest a system design converter that generates computer statements from application definition data. The Examiner refers to col. 4, line 58 – col. 5, line 9 of Gloudeman as providing this teaching. This is also the same portion of Gloudeman that the Examiner asserts discloses the data provider interface. As described above, this portion of Gloudeman describes the conversion of third party database access methods to the APIs used by the four databases of Gloudeman. It does not teach or suggest the conversion of application definition data. Gloudeman only teaches the generation of the database access commands and responses by the third party layer 56. Gloudeman fails to teach or suggest the generation of computer statements that implement control logic for an application program. Thus, the third party layer 56 is not a system design converter, does not accept application definition data, and does not generate computer statements that implement control logic described by the application definition data. The absence of these elements from the Gloudeman reference prevents the reference from anticipating claims 1-3, 11-13, 15-17, and 25-27.

C. The Gloudeman reference does not teach, explicitly or impliedly, a computer tool interface coupled to the system design converter for providing the system design converter with data from the database through the data provider interface. As noted above, the Gloudeman reference fails to teach a data provider and a system design converter as set forth in the rejected claims. Thus, it cannot disclose a computer tool interface coupled to a system design converter. Additionally, the computer tool interface provides data to the system design converter for the generation of the computer statements. *See Specification*, page 21, line 6 to page 22, line 10. The claims also require that the computer tool interface provide the system design converter with some of the data for the generation of the computer statements from the database through the data provider interface.

The Gloudeman reference does not teach a component that provides data to another component for the generation of computer statements that implement control logic for an application program. The Examiner again relies upon col. 4, line 58 to col. 5, line 9 of Gloudeman for this teaching. As noted above, this portion of Gloudeman does not teach or suggest a component that generates computer statements that implement control logic for an application as the system design converter of the claimed invention. Hence, the system of Gloudeman does not have a need for a computer tool interface that accesses a database through the data provider interface, *Specification*, page 21, lines 13-15, so the system design converter can generate computer statements for an application solution with data from the database, *Specification*, page 22, lines 4-6. The absence of the computer tool interface from Gloudeman is further support that claims 1-3, 11-13, 15-17, and 25-27 are not anticipated by Gloudeman.

D. The Gloudeman reference does not teach, explicitly or impliedly, an external program module interface coupled to the system design converter to provide the system design converter with external program modules. The claimed external program module interface enables the system design converter to receive external program modules for the generation of an application solution. *Specification*, page 22, lines 4-10.

The Examiner relies upon col. 3, line 60 to col. 4, line 11 in Gloudeman for this teaching. This portion of Gloudeman, however, relates to the integration of an external system to the system of Gloudeman through the third party interface 56 described above. It does not describe the accessing of program modules for generation of computer statements in an application solution, but rather an avenue for the migration of data from an external system to the databases of the Gloudeman system. The claimed invention distinguishes between data received through the computer tool interface and the external programs received through the external program interface. No such distinction is taught by Gloudeman because Gloudeman does not teach the receipt of external programs through an external program module interface. The absence of an external program interface in Gloudeman renders Gloudeman ineffective for supporting a 102 ground of rejection for claims 1-3, 11-13, 15-17, and 25-27.

The failure of Gloudeman to disclose even one of these elements is sufficient to overrule the section 102 ground of rejection asserted by the Examiner. The failure of Gloudeman to disclose any of these elements is evidence that Gloudeman is severely deficient and cannot support the Examiner's ground of rejection.

CLAIMS 5 and 19

Claims 5 and 19 are separately patentable because they require the external program module interface to include common components for application support. Common components are contained in a software library and are used by the system design converter to generate computer statements for implementing an application solution. *Specification*, page 32, lines 1-18. They are provided to the system design converter through the external program module interface. *Specification*, page 21, line 20 to page 22, line 1.

The Gloudeman reference does not teach the inclusion of common components so they may be provided to a system design converter through an external program module interface. The Examiner relies upon col. 4, lines 25-57 in Gloudeman for this teaching. This section discloses decoupled components and applications. The Gloudeman reference, however, does not disclose the generation of the computer statements that implement the application nor does it teach that the decoupled components are used by a system design converter to generate computer statements for an application solution. Every aspect of the claimed invention must be disclosed in the reference for a proper anticipation rejection. Because the Gloudeman reference fails to teach the provision of application support components to a system design converter through an external program module interface, it cannot properly support an anticipation rejection of claims 5 and 19.

CLAIMS 6 and 20

Claims 6 and 20 are separately patentable because they require the external program module interface to include Web-based components for coupling the computer statements generated by the system design converter to another application over the Internet. Web-based components are used for coupling application solutions executing computer statements generated by the system design converter to the Internet for communication purposes. *Specification*, page 10, lines 11-18. They are provided to the system design converter through the external program module interface. *Specification*, page 21, line 20 to page 22, line 1.

The Gloudeman reference does not teach the inclusion of Web-based components so they couple an application solution to the Internet. The Examiner relies upon col. 4, lines 12-24 in Gloudeman for this teaching. This section discloses, however, that a user may be coupled to the building automation system through the Internet. Nowhere does it teach that components supporting communication with a user over the Internet so a user can view data through the information layer may also be used to couple the application layers 78 and 80 to the Internet. Claims 6 and 20 require the system design converter to use web-based components accessed through the external program module interface for coupling the application solution executing the computer statements generated by the system converter to the Internet. The Gloudeman reference, fails to disclose the access of web-based components through an external program module so application solutions can be coupled to the Internet. Therefore, the Gloudeman reference cannot properly anticipate claims 6 and 20.

CLAIMS 7-8 and 21-22

Claims 7-8 and 21-22 are separately patentable because they require the common components to include operating system communication components for coupling the application solutions to another application through the operating system. Operating system components are used for coupling application solutions executing computer statements generated by the system design converter to other applications through the operating system. *Specification*, page 10, lines 18-23, and page 19, lines 11-17. They are provided to the system design converter through the external program module interface. *Specification*, page 21, line 20 to page 22, line 1.

The Gloudeman reference does not teach the inclusion of operating system components for the purpose of coupling an application solution to another application through the operating system. The Examiner relies upon col. 4, lines 12-24 and lines 51-50 in Gloudeman for this teaching. Nowhere does Gloudeman teach that the applications can be coupled to other applications through the operating system. If any such communication can occur in the system of Gloudeman, it can only occur because the applications have been programmed with the requisite instructions to coordinate with the operating system. In Applicant's system, the application designer need not be proficient in operating system procedures for such communication. *Specification*, page 10, lines 20-23. Instead, the operating system components are accessed by the system design converter through the external program module interface for coupling the application solution executing the computer statements generated by the system converter to another application through the operating system. The Gloudeman reference fails to disclose the access of operating system components through an external program module so application solutions can be coupled to other applications through an operating system. Therefore, the Gloudeman reference cannot properly anticipate claims 7-8 and 21-22.

CLAIMS 9 and 23

Claims 9 and 23 are separately patentable because they require the Web-based components to couple the computer statements of the application solution generated by the system design converter to another application over the Internet through a customer web portal. This limitation is directed to the use of a customer web portal to centralize access for applications and users for various purposes. *Specification*, page 11, lines 8-19, and page 15, lines 6-16. The customer web portal is a preferred method of communication access over the Internet. *Specification*, page 19, lines 8-11.

The Examiner relies upon col. 4, lines 12-24 in Gloudeman for this teaching. The only type of communication connection discussed in this portion of Gloudeman is a generic browser. A browser is not a customer web portal. As set forth in specification sections cited above, the customer web portal provides a central site for posting application data, applications, and reports that may be used to develop applications, verify applications, and add applications to a building system. The Gloudeman reference is devoid of teachings on any type of web portal. Therefore, the Gloudeman reference does not anticipate claims 9 and 23.

CLAIMS 10 and 24

Claims 10 and 24 are separately patentable because they require a configuration utility for developing a file structure representative of a building system. The claims also require the configuration utility to associate configuration data with components identified in the file structure. *Specification*, page 11, line 20 – page 12, line 5. The configuration data files generated by the configuration utility enable applications to obtain data for building components. *Specification*, page 15, line 17 – page 16, line 2. These configuration data files generated by the configuration utility are accessed through the data provider. *Specification*, page 29, line 19 to page 30, line 15; FIG. 4.

The Examiner relies upon col. 5, lines 27-34 in Gloudeman for this teaching. This section refers to a building system data store and a naming server implemented in the data store. The building system data store is not a configuration utility, but rather a data repository. The naming server is described at col. 5, line 62 to col. 6, line 60 of Gloudeman. The purpose of the naming server is to implement a naming scheme that allows distributed objects to be located anywhere across the network. *Gloudeman*, col. 5, lines 53-65. These objects perform a function, which may be used by a system developer to name the object with the naming server. *Gloudeman*, col. 5, line 65 to col. 6, line 1; col. 6, lines 19-23. The naming server, therefore, does not develop a file structure representative of a building system nor does it associate configuration data with components identified in the file structure as required by claims 10 and 24. Instead, it simply generates unique names for objects.

Gloudeman also discloses that the optimization application layer updates attributes stored in the building system data store. Again, no teaching is provided that the optimization application layer develops the file structure or that it associates configuration data with components identified in the file structure. Because, Gloudeman is silent regarding the development of the files in the building system data store and the structure of the files in that data store, it cannot anticipate claims 10 and 24.

CLAIMS 14 and 28

Claims 14 and 28 are separately patentable because they require a scheduling service for activating the data collector interface to interrogate the external data sources for data to be stored in the database. *Specification*, page 12, line 21 – page 13, line 2; page 16, lines 3-9; page 27, lines 17-19; FIG. 3. The data obtained from these external data sources are used to update databases in the building system.

The Examiner relies upon col. 4, line 58 to col. 5, line 9; and col. 5, lines 39-61 in Gloudeman for this teaching. The first cited section has been discussed above. It is directed to a user interface to the databases through a standard communication and database API layer or a third party interface layer. There is no teaching in this section that a scheduling service activates either layer to interrogate external data sources for data that are used to update a database. The other cited section refers to two application layers. The real-time and the optimization layers of Gloudeman include applications that may execute periodically either to manage operation of the facility automatically or to adjust the system operating attributes automatically. Nothing in this section discloses that these application layers are activated by a scheduling service for the purpose of interrogating external data sources for data to be stored in a database.. Consequently, Gloudeman fails to anticipate claims 14 and 28.

II. Rejection under 35 USC 103(a) Over U.S. Patent No. 6,141,595 In View Of Pub. No. 2003/0229652

CLAIMS 4 and 18

In the final Office Action, the Examiner asserted that claims 4 and 18 were obvious over Gloudeman in view of Bakalash. The Examiner, however, failed to specify how this combination taught or suggested a data mart that is configured in *one* of a snowflake and star data organization. That is, these claims require that the data mart be in one or the other configuration. *Specification*, page 10, lines 3-7; page 22, lines 12-16. These claims require that the data mart have one of two particular configurations. The real-time database of claims 3 and 17, which are relevant to claims 4 and 18 through dependency, may have a different structure and interface. For example, the data mart may use either a star or snowflake data organization, which facilitates data mining, while the real-time database may retain a more conventional Standard Query Language (SQL) interface. *Specification*, page 22, lines 11-20. Nothing is available in either Gloudeman or Bakalash that would suggest configuring some of the databases in Gloudeman in a star schema with snowflake extensions while other databases are configured in some other manner.

The Examiner relies upon Gloudeman for the rejection of claims 3 and 17 from which claims 4 and 18 depend, respectively. The failure of Gloudeman to anticipate those claims has been discussed above. Additionally, Applicant notes that Bakalash does not teach a database being configured in either a star or a snowflake configuration. Instead, Bakalash teaches a star schema with snowflake extensions. *Bakalash*, ¶ 73. Thus, Bakalash does not teach a data mart being in either a star or snowflake configuration, but rather as being configured in a star schema with snowflake extensions. The combination, even if appropriate, does not teach or suggest the claimed data mart, which is separate from the real-time database in claims 3 and 17, being configured in one of a star or snowflake configuration. Moreover, Bakalash does not teach or suggest that a system design converter access a star schema database with snowflake extensions through a computer tool interface and data provider.

Additionally, the Examiner has failed to show any motivation for combining the star schema database having snowflake extensions with the system of Gloudeman. The system of Gloudeman includes a plurality of databases that are sandwiched between a building system interface and a layer of system database APIs. No motivation is given as to why one of ordinary skill in the art would insert the database structure of Bakalash between these two different layers of Gloudeman.

Because the Examiner has failed to show motivation to combine a database in a star schema having snowflake extensions with the object architecture of Gloudeman, the obviousness rejection of claims 4 and 18 should be overruled. Additionally, even if these references were properly combined, they do not teach or suggest a data mart having either a star or snowflake configuration and a real-time database with a data provider and computer tool interface for providing a system design converter access to data in the data mart and real-time database.

CONCLUSION

As set forth above, the Gloudeman reference fails to anticipate claims 1-3, 5-17, and 19-28. Additionally, one of ordinary skill in the art would not be motivated to combine Gloudeman and Bakalash because neither reference suggests the sandwiching of a star schema database having snowflake extensions between a building system layer and a layer of system database APIs. Moreover, the combination of these two references does not teach or suggest a data mart having either a star or snowflake configuration and a real-time database with a data provider and computer tool interface for providing a system design converter access to data in the data mart and real-time database. Thus, claims 4 and 18 are not rendered obvious by the combination of Gloudeman with Bakalash. Therefore, the Board of Appeals, is respectfully requested to reverse the rejection of pending claims 1-28.

Respectfully submitted,
MAGINOT, MOORE & BECK LLP

David M. Lockman
Attorney for Applicant
Registration No. 34,214

February 22, 2007
Maginot, Moore & Beck LLP
Chase Tower
111 Monument Circle, Suite 3250
Indianapolis, Indiana 46204-5109
(317) 638-2922 Telephone
(317) 638-2139 Facsimile

(8) CLAIMS APPENDIX

Listing of Claims

1. A system for a building system application comprising:

a database;

a data provider interface for converting between a common database access method and a database application programming interface (API); and

an application infrastructure, the infrastructure comprising:

a system design converter for converting application definition data into computer statements for implementing control logic of application definition data;

a computer tool interface coupled to the system design converter, the computer tool interface providing the system design converter with data from the database through the data provider interface;

an external program module interface coupled to the system design converter, the external program module interface providing the system design converter with external program modules; and

the system design converter includes data obtained through the computer tool interface and external program modules obtained through the external program module interface with the computer statements for implementing control logic of application definition data to generate a building system application.

2. The system of claim 1 wherein the database is comprised of a plurality of databases.

3. The system of claim 2, the database being comprised of a real-time database and a data mart.

4. The system of claim 3, the data mart being configured in one of a snowflake and star data organization.

5. The system of claim 1, the external program module interface further comprising:

common components for application support.

6. The system of claim 1, the external program module interface comprising:

Web-based components for coupling the computer statements for implementing the control logic of the application definition data to another application over the Internet.

7. The system of claim 5, the common components further comprising:

operating system communication components for coupling the computer statements for implementing the control logic of the application definition data to another application through an operating system.

8. The system of claim 7 wherein the operating system communication components communicate through a Windows operating system.

9. The system of claim 6 wherein the Web-based components couple the computer statements for implementing the control logic of the application definition data to another application over the Internet through a customer web portal.

10. The system of claim 1 further comprising:

a configuration utility for developing a file structure representative of a building system and for associating configuration data with components identified in the file structure.

11. The system of claim 1 further comprising:

a data collector interface for coupling external data sources to the database.

12. The system of claim 11 wherein the data collector interface converts data from the native format for an external data source to one that is compatible with the database structure.

13. The system of claim 12 further comprising:

transaction services for generating instructions for the database API to store the converted data in the database.

14. The system of claim 11 further comprising:

a scheduling service for activating the data collector interface to interrogate the external data sources for data to be stored in the database.

15. A method for supporting a building system application comprising:

storing data in a database;

converting application definition data into computer statements for implementing control logic of application definition data;

converting between common database access method instructions in the computer statements and database application programming interface (API) instructions for the database so that the common database access method instructions may access the database; and

using data obtained from the database with the computer statements and incorporating external program modules in the computer statements to generate building system applications.

16. The method of claim 15 wherein the storing of data in the database includes storing the data in a plurality of databases within the database.

17. The method of claim 15 wherein the storing of data in the database includes storing the data in one of a real-time database and a data mart.

18. The method of claim 17 further comprising:

configuring the data mart in one of a snowflake and star data organization.

19. The method of claim 15 further comprising:

coupling common components to the computer statements for implementing control logic of application definition data for communication support.

20. The method of claim 19, the common component coupling comprising:

coupling the computer statements for implementing control logic of application definition data to another application through a Web-based component for communication over the Internet.

21. The method of claim 19, the common component coupling comprising:

coupling the computer statements for implementing control logic of application definition data to another application through an operating system communication component for supporting application communication through the operating system.

22. The method of claim 21 wherein the operating system common component coupling includes coupling a Window-based communication component to the computer statements for implementing control logic of application definition data.

23. The method of claim 20 wherein the communication through the Web-based component over the Internet is through a customer web portal.

24. The method of claim 15 further comprising:

developing a file structure having components representative of a building system; and

associating configuration data with the components identified in the file structure.

25. The method of claim 15 further comprising:

coupling external data sources to the database.

26. The method of claim 25 further comprising:

converting data from a native format for an external data source to one that is compatible with the database.

27. The method of claim 26 further comprising:

generating instructions for the database API to store the converted data in the database.

28. The method of claim 27 further comprising:

interrogating, on a scheduled basis, a plurality of external data sources for data to be stored in the database.

(Withdrawn) Claims 29-55.

(9) EVIDENCE APPENDIX

No evidence was submitted under rules 1.130, 1.131, or 1.132. Additionally, no other evidence has been entered in the record by the Examiner upon which the Applicant relies.

(10) RELATED PROCEEDINGS APPENDIX

No proceedings were identified in the Related Appeals and Interferences presented above. Therefore, no decisions of a court or the Board are contained herein.